AIAA 200-4373

NONLINEAR CONTROL OF MULTIPLE UAVS IN CLOSE-COUPLED FORMATION FLIGHT

Corey Schumacher*
Air Force Research Laboratory (AFRL/VACA)
2210 Eighth St, Bldg. 146, Sutie 21
Wright-Patterson AFB, OH 45433-7531†

Sahjendra N. Singh
ECE, University of Nevada
Las Vegas, NV 89154-4026

ABSTRACT

This paper studies the control of multiple UAV's flying in a close-coupled formation for the purposes of drag reduction. A controller design methodology for use in a two-UAV formation is presented. The outer loop controller is based on sliding mode control in a modified wind-axis coordinate system. The adaptive dynamic inversion inner-loop tracks these input commands using incomplete knowledge of the aircraft dynamics. The controller is tested in a two-vehicle formation flight simulation. Excellent command tracking and performance are achieved without use of specific knowledge of the formation flight effects in the inner-loop dynamic inversion. Simulation results demonstrate that the proposed controller design enables the formation to follow commanded trajectories. The controller also enables the wing UAV to maneuver in the lead UAV's wake, and to hold a desired position in the lead vehicle's wake. The induced lift on the wing UAV due to formation flight is shown to cause reduced angle-of-attack and thrust for the wing vehicle.

1. Introduction

It is known that aircraft with large aspect ratio wings have better overall aerodynamic efficiency because of reduction in induced drag for a given lift. However large aspect ratio implies large wingspan for a given area. This means for lightweight design the resulting structure will be unreasonably flexible and fragile. As a feasible alternative the similar kind of improvement in overall efficiency can be achieved by flying many similar reasonable aspect ratio wing aircraft in close formation. In an idealized case of n identical aircraft, each with aspect ratio a, flying in wing-tip to wing-tip formation the effect would be that of a single craft with $n.a$ aspect ratio. This kind of drag reduction in close formation is due to favorable wake-vortex encounters. Wind tunnel tests have shown that this drag benefit is real and analytical studies predict that the benefit increases as additional aircraft are added to the formation. Flying in close-coupled formation refers to those formation flights where wake vortex encounters are intentional.

A number of recent studies have examined the problem of close-coupled formation flight for multiple aircraft. The formation control of linear aircraft models has been considered [1-3]. Formation control of two aircraft using PID feedback has been presented in [1], where it is assumed that Mach-, altitude-, and heading-hold autopilots are available, and only the outer-loop problem is examined. A linear decentralized controller is presented in [2]. In [3], an optimal peak-seeking controller is presented, which attempts to find the optimal location for minimum drag during aircraft flight. These studies have all been limited to linear aircraft models.

The modeling of vortex-effects on the aircraft in close-coupled formation flight is also a subject of considerable recent interest, and is very important for effective control system design. These aerodynamic coupling effects are highly nonlinear and difficult to model accurately. The results of wind tunnel experiments to measure forces and moments on a trail aircraft in a vortex are presented in [4].

This paper examines the controller design problem for an unmanned aerial vehicle (UAV) flying in the wake of another, identical UAV for the purposes of drag reduction. This is a particularly challenging control problem for a number of reasons. The vortex generated by the lead vehicle induces large, nonlinear forces and moments on the trail (or chase) vehicle. The induced rolling moment, in particular, is quite large and highly nonlinear, and varies substantially in both magnitude and sign as the trail aircraft changes position in the vortex. Additionally, the formation flight effects are very difficult to measure and to model. Detailed accurate knowledge of the formation flight effects is generally not available during flight.

The controller developed here is a two-loop controller. The outer-loop is a sliding mode controller that maintains the trail vehicle's relative position behind the lead vehicle, even during lead vehicle maneuvers. The outer-loop controller generates body axis rate

* Research Aerospace Engineer; Member, AIAA

commands for use in the inner loop. This outer-loop control methodology is developed in [5]. This paper takes the outer-loop controller developed in [5] and applies it to a full vehicle model with inner-loop dynamics. The inner-loop controller is a dynamic inversion controller with an adaptive neural network that takes body axis p, q, r, commands as inputs and generates control surface deflection commands. The adaptive dynamic inversion method used in the inner loop is developed in detailed in [6-9].

This paper presents an effective control design method for close-coupled formation flight. Simulation results show that the controller achieves good tracking of relative position commands. The adaptive dynamic inversion inner-loop tracks the p-, q-, r-commands very well, even though it uses only very limited information about the system dynamics.

The organization of this paper is as follows. Section 2 provides details of the formation flight model used in the simulation. In Section 3, the formation flight control system is discussed, and simulation results are presented in Section 4.

2. Formation Flight Modeling

A six-degree of freedom nonlinear simulation model has been developed in Matlab/Simulink. The nonlinear equations of motion for an aircraft were used, but the constant. This restriction does not greatly reduce the validity of the simulation results because only relatively small maneuvers (with approximately constant aerodynamic nominal (not including formation effects) aerodynamic coefficients were assumed to be coefficients) are performed in the simulation results presented. A table look-up for the formation flight data was also included, containing the changes in all force and moment coefficients as functions of relative X, Y, and Z separation between the lead and chase vehicles. This model was generated for 1-g straight and level flight trimmed at Mach 0.8 and altitude 45000 ft. The relative position of the lead vehicle with respect to the chase vehicles is expressed in an inertial, right-handed reference frame with the X-axis out the lead aircraft's nose, the Y-axis out the right wing, and the Z-axis down, as shown in Figure 1.

The formation flight data were given in the following form:

$$C_{L,i} = C_{L,i}(s) + \sum_{j,i\neq j}\left(\frac{\Delta C_{L,i}}{C_{L,j}}\right)C_{L,j}$$

$$C_{D,i} = C_{D,i}(s) + \sum_{j,i\neq j}\left(\frac{\Delta C_{D,i}}{C_{L,i}C_{L,j}}\right)C_{L,i}C_{L,j}$$

$$.C_{m,i} = C_{m,i}(s) + \sum_{j,i\neq j}\left(\frac{\Delta C_{m,i}}{C_{L,j}}\right)C_{L,j}$$

$$C_{l,i} = C_{l,i}(s) + \sum_{j,i\neq j}\left(\frac{\Delta C_{l,i}}{C_{L,j}}\right)C_{L,j}$$

$$C_{n,i} = C_{n,i}(s) + \sum_{j,i\neq j}\left(\frac{\Delta C_{n,i}}{C_{L,i}C_{L,j}}\right)C_{L,i}C_{L,j}$$

$$C_{Y,i} = C_{Y,i}(s) + \sum_{j,i\neq j}\left(\frac{\Delta C_{Y,i}}{C_{L,i}C_{L,j}}\right)C_{L,i}C_{L,j} \qquad (1)\text{-}(6)$$

where the terms in the parenthesis are provided in table look-up as a function of relative distances. It should be noted that the coupling effect is given as an add-on to six unperturbed aerodynamic forces and moments.

This model provides a great deal of information about the vortex effects during close-coupled formation flight, but extensive modeling work is still required. Time delay effects and the effects of aircraft rolling motion, as well as other, smaller, effects still need to be included.

The primary effects of the aerodynamic coupling are on drag, lift, and rolling moment experienced by the trail aircraft. The lead aircraft does experience some aerodynamic coupling, but it is relatively minor. The formation flight effects are highly nonlinear, and it is extremely difficult to completely model the formation flight effects.

3. Formation Flight Control

The formation flight controller presented in this paper controls both the lead and wing vehicles in a two-vehicle formation. The lead vehicle tracks commands in velocity, heading angle and flight path angle. The wing, or trail, vehicle follows the lead vehicle's maneuvers, and, additionally, tracks commands in desired relative separation between the lead and wing vehicles. The formation controller is composed of two major parts: an outer-loop sliding mode controller and an inner-loop adaptive dynamic inversion controller. The outer loop maintains relative x, y, and z position of the trail vehicle with respect to the lead vehicle in the formation, and generates Power Lever Angle (PLA) commands and body axis rotation rate commands (p,q,r commands). The p,q,r commands are input to the inner-loop dynamic inversion control, which generates control surface deflections to achieve the desired body axis rotation rates.

3.1 Lead UAV Controller

The lead UAV controller is composed of two parts: a sliding mode outer-loop controller and a dynamic inversion inner-loop controller. The basic structure of the lead vehicle controller is shown in Figure 1.

Outer Loop – Variable Structure Control

The outer loop controller is a sliding mode design, taking V, χ, γ commands as inputs and using body rates p,q,r as control inputs to achieve the desired trajectory. The controller also must maintain stability. The controller is based on the method developed in [5]. It is variable structure controller derived in a simplified wind coordinate system obtained from the wind coordinate system for which the velocity roll is zero. The switching surface is given by:

$$s = \begin{pmatrix} e_{10} + p_1 \int_0^t e_{10} dt \\ \dot{e}_{20} + 2\zeta\omega_\chi e_{20} + \omega_\chi^2 \int_0^t e_{20} dt \\ \dot{e}_{30} + 2\zeta\omega_\gamma e_{30} + \omega_\gamma^2 \int_0^t e_{30} dt \end{pmatrix} \tag{7}$$

where p_1, ζ, ω_χ and ω_γ are positive numbers, and $e_0 = (e_{10}, e_{20}, e_{30})^T = (V_c - V, \chi c - \chi, \gamma c - \gamma)^T$ is the trajectory tracking error. During the sliding phase the error e0 converges to zero. The switching surface includes integral error feedback for robustness.

Differentiating s along the trajectory gives

$$\dot{s} = a^* + B^* u_l \tag{8}$$

and solving for the equivalent control so that $\dot{V}$ is negative definite. This results in a control law of the form

$$u_l = \left(B^*\right)^{-1}\left(-a^* - c_1 s - K sat(s)\right) \tag{9}$$

The final outputs of the outer-loop sliding mode controller u are commands in thrust, $\dot{L}$, and $\dot{\mu}$. These commands must then be converted to body-axis p,q,r commands which are sent to the inner-loop dynamic inversion controller. Further details of the outer-loop controller can be found in [5].

Conversion of Control Commands to Body Rates

The lift-rate command generated by the outer-loop sliding mode controller is converted to a body-axis rate command q_c. In the simulation, lift is calculated as $L = \bar{q}SC_L$ where $\bar{q}$ is the dynamic pressure, S is the reference area, and the lift coefficient C_L is:

$$C_L = C_{L0} + C_{L\alpha}\alpha + C_{LV}V + control\ terms \tag{10}$$

We assume that the effect of the control surface deflections on lift is small and neglect those terms. We also assume that velocity is approximately constant during the time scale of a pitch maneuver, and neglect the V terms when taking the derivative. Then $\dot{L}$ becomes $\dot{L} = \bar{q}SC_{L\alpha}\dot{\alpha}$. After substituting $\dot{L}_c$ for $\dot{L}$ this equation can be solved for q_c, which is input to the inner loop. It is also assumed that sideslip angle is small enough for the $p_s \tan\beta$ can be neglected in calculating the pitch command.

The $\dot{\mu}$ command generated by the outer loop is converted to body-axis p and r commands. If $\dot{\mu} = f_\mu(x,p,r)$ and $\dot{\beta} = f_\beta(x,p,r)$, p and r commands can be calculated by setting $\dot{\mu}_c = f_\mu(x,p_c,r_c)$ This is done by solving $\dot{\mu}_c = \dot{\mu}(p_c, r_c)$ and $\dot{\beta}_c = -K\beta = \dot{\beta}(x,p_c,r_c)$ simultaneously for p_c and r_c. The value K=10 is chosen to provide a stable sideslip response which prevents the growth of sideslip angle during maneuvers. The equations for p_c, q_c, r_c become:

$$q_c = \frac{\dot{L}_c}{\bar{q}SC_{L\alpha}} + \frac{\bar{q}SC_L}{mV\cos\beta} + \frac{T_x\sin\alpha - T_z\cos\alpha}{mV\cos\beta} + \frac{g(\cos\alpha\cos\phi\cos\theta + \sin\alpha\sin\theta)}{V\cos\beta} \tag{11}$$

$$p_c = \mu_c\cos\alpha\cos\beta - (f_\beta + K)\sin\alpha - \dot{\gamma}\cos\mu\sin\beta\cos\alpha - \dot{\chi}\cos\alpha\cos\beta(\sin\alpha + \cos\gamma\sin\mu\tan\beta)$$

$$r_c = p_c\tan\beta + (f_\beta + K)\sec\alpha$$

In the above equation, K=10 and f_β is defined by $\dot{\beta} = -r_s + f_\beta$. That is, f_β is composed of all of the elements of the $\dot{\beta}$ equation that are not functions of p or r.

Inner Loop – Dynamic Inversion

The inner-loop for the lead vehicle is a dynamic inversion controller that takes body-axis rate ***p,q,r*** commands as inputs and generates control surface deflection commands as outputs. In dynamic inversion control, we seek to linearize a nonlinear system [9]. The aircraft's dynamics can be put in the form

$$\dot{\mathbf{x}} = \mathrm{f}(\mathbf{x}) + \mathrm{g}(\mathbf{x})\mathbf{u}, \tag{12}$$

where **x** is a vector the controlled states and **u** is a vector of controls and f(**x**) and/or g(**x**) is nonlinear. The number of controlled states and controls need to be the same (a square system). The control can be calculated according to

$$\mathbf{u}_c = \mathrm{g}^{-1}(\mathbf{x})(\mathbf{u}_d - \mathrm{f}(\mathbf{x})), \tag{13}$$

where $\mathbf{u}_d$ is the desired response of $\dot{\mathbf{x}}$. Substituting (12) into (12), for **u**, results in

$$\dot{\mathbf{x}}_d = \mathbf{u}_d \tag{14}$$

and any nonlinearities in f(**x**) and g(**x**) are cancelled. In the figure, the body rotation-rate commands from the outer loop are the inputs. In this application, we use an implicit command-following controller with proportional-integral desired dynamics, so the desired dynamics $\mathbf{u}_d$ become:

$$\mathbf{u}_d = \tfrac{1}{2}K_b\mathbf{x}_c - K_b\mathbf{x} + \tfrac{1}{4}K_b^2\int(\mathbf{x}_c - \mathbf{x})dt. \tag{15}$$

For this application we chose K_b = 10. The desired dynamics are identical in all three channels, roll, pitch, and yaw. Control surface deflection commands are then calculated according to (13). In the simulation results presented later, full knowledge of the vehicle dynamics is used in calculating the dynamic inversion, except for the forces and moments generated by the formation flight effects.

3.2 Wing UAV Controller

The wing, or trail, UAV controller has a somewhat more complicated task than that of the lead vehicle. The wing UAV's controller must follow the lead vehicle's maneuvers while also maintaining the proper relative position to take advantage of the lead's vortex for drag reduction. The basic structure of the controller for the wing vehicle is similar to that used for the lead, with an outer-loop sliding mode controller derived in a modified wind-axis system with the velocity roll angle set to zero. The inner-loop is a dynamic inversion controller, but for the wing vehicle an adaptive neural network is added to improve the vehicle response. Knowledge of the formation flight effects is not used in the dynamic inversion controller, and the formation effects are much larger on the wing than on the lead vehicle. The adaptive neural network is used to compensate for the disturbance moments generated by the lead UAV's vortex.

Outer-Loop Wing Sliding Mode Controller

The wing UAV outer-loop controller receives position separation trajectory commands that it must track. These trajectories are received from 3rd order command generators which convert relative x, y, z separation commands into smooth reference trajectories which can be tracked by the wing UAV's controller.

For the derivation of the variable structure control law, we select a switching surface given by

$$\dot{s} = \dot{e} + 2\zeta\omega_n e + \omega_n x_s = 0 \qquad (16)$$

where ζ and ω_n are positive numbers, $e=p-p_c=(e_1,e_2,e_3)^T$ is the trajectory tracking error, $p=(x,y,z)^T$ is the relative position of the lead aircraft with respect to the trail aircraft in the wind axis frame, $p_c = (x_c,y_c,z_c)T$ is the vector reference trajectory to be tracked by p, and x_s is the integral of the tracking error, that is

$$\dot{x}_s = e \qquad (7)$$

The switching surface include integral error feedback for robustness. The control law is derived using a Lyapunov function of the form

$$W = \sum_{k=1}^{3} |s_k| \qquad (18)$$

and the resulting control law is of the form

$$u_w = \left(B^*\right)^{-1}\left(-a^* - c_1 s - K sat(s)\right) \qquad (19)$$

The wing controller as designed achieves separation of the centers of mass of the vehicles. A modification to the separation commands is required to maintain the proper relative wingtip positions. If the desired z-separation is zero, and the desired y-separation is y_f, then the modified relative position commands become

$$\begin{aligned} y_c &= 0.5 y_f \left(\cos\mu_l + \cos\mu_w\right) \\ z_c &= 0.5 y_f \left(\sin\mu_l + \sin\mu_w\right) \end{aligned} \qquad (20)$$

where μ_l and μ_w are the lead and wing vehicle velocity roll angles respectively. Inner-loop body-axis rate commands p_c, q_c, r_c are calculated from the $\dot{L}$, $\dot{\mu}$ commands in a manner identical to that used for the lead vehicle. A much more detailed description of the outer-loop control law for both the lead and wing UAVs can be found in [5].

Inner-Loop Adaptive Dynamic Inversion Controller

The basic, inner-loop dynamic inversion controller for the wing vehicle is largely similar to that used in the lead vehicle, with a few modifications so that it can be augmented with an adaptive neural network. The primary change in the nominal dynamic inversion is in the desired dynamics calculations. The desired dynamics calculated using (15) are augmented with the output of a second-order command-filter, resulting in:

$$\begin{aligned} \mathbf{u}_d = {} & \tfrac{1}{2} K_b \mathbf{x}_c - K_b \mathbf{x} + \tfrac{1}{4} K_b^2 \int \left(\mathbf{x}_c - \mathbf{x}\right) dt \\ & + K_b x_c + \dot{x}_c \end{aligned} \qquad (21)$$

The last two terms are added to modify the resulting error dynamics and allow the computation of a stable weight-adaptation law for the neural network using a Lyapunov function. The modification of the desired dynamics u_d and the derivation of the weight update law are discussed in detail in [9].

Adaptive Neural Network

In this study we use a linear-in-parameters neural network. An identical network structure was used in all three channels, with only a slight variation in the inputs. The inputs to each network were the vehicle's body axis rates p, q, and r, the squashed control for that control channel, $\sigma(u_{di})$, where $\sigma(u_{d1}) = \frac{2}{1+e^{-0.1 u_{d1}}} - 1$. The purpose of the squashing function is to guarantee that the control-related input to the neural network has a magnitude less than one, guaranteeing a solution to the algebraic loop around the network. The basis functions in each network were linear in the normalized inputs, without cross terms. Thus, the basis functions used were *1, p, q, r,* Δy, Δz, $\sigma(u_i)$, resulting in seven weights and seven basis functions in each of the three networks.

Weight-Update Law

An important part of any neural network is the rule governing its adaptation. In the method used here, the weight adaptation law is derived via Lyapunov stability analysis, and is dependent on the desired dynamics in the dynamic inversion. In [9], it is shown that the appropriate weight-update law for use with a dynamic inversion controller with desired dynamics given by (18) is

$$\dot{w}_i = -\gamma_i\left(\left(\frac{1}{2K_i}y + \frac{1+K_i}{2K_pK_i}\dot{y}\right)G_i + \eta_i|\vec{y}|w_i\right) \qquad (22)$$

where w_i is the i[th] weight of the linear-in-parameters neural network, $\vec{y} = [y, \dot{y}]^T$, and $\gamma \in \Re^1, \eta \in \Re^2$ γ_i is the learning rate, b_i is the i[th] basis function, and η is an *e-modification* factor to unsure that **w** is bounded. In this application, the gains were chosen as γ_i=50 and η = [0.05 0.05]. Also, $K_i = 0.25K_b^2$ and $K_p = K_b$. Finally, $y = \int(x_c - x)dt$ and $\dot{y} = x_c - x$. This neural network results in an adaptive controller, with no long-term memory. No off-line learning is required with this adaptive network, because it has no long-term memory. With the weight update law given by (19), closed-loop stability and boundedness of the control signals are guaranteed.

The output of the network is formed by $u_{ad} = b^T w$, where b is the vector of basis functions and w is the vector of weights. Ideally, the output of the network will exactly match the inversion error, resulting in exact dynamic inversion. This output is subtracted from u_d in (15) to compensate for the inversion error. Then the dynamic response becomes

$$\dot{x} = \tfrac{1}{2}K_b\mathbf{x}_c - K_b\mathbf{x} + \tfrac{1}{4}K_b^2\int(\mathbf{x}_c - \mathbf{x})dt \,. \\ +K_b x_c + \dot{x}_c + \Delta - u_{ad} \qquad (23)$$

where Δ is the inversion error. If u_{ad} = Δ, then the network exactly matches the inversion error and the desired dynamics are achieved exactly.

4. Simulation Results

The multiloop controller detailed in the previous sections was tested in the formation flight simulation with two UAVs flying in formation at mach 0.8 and an altitude of 45,000 ft. The simulation was initialized with both UAV's in a trim condition, with the trail UAV 80 feet behind and 33.75 feet to the side of the lead vehicle. The relative z-separation is initially zero. The x-position was chosen so that there would be adequate longitudinal separation between the two vehicles for safety purposes. The y-position was chosen to be 90% of the aircraft wingspan, which is near the optimal position for drag reduction. The trail UAV is approximately but not perfectly trimmed in the lead UAV's wake.

The commanded maneuver for which simulation results are shown is a complex one. Simultaneously, step commands are given to the lead vehicle in flight speed, flight path angle, and heading angle. Additionally, relative position command changes are given to the trail vehicle at the same time.

The lead aircraft is commanded to change flight speed from 774.44 ft/s to 772.44 ft/s, flight path angle from 0 deg to 3 deg, and heading angle from 0 deg to 5 deg. The wing vehicle is commanded to change the x-separation form 80 ft to 75 ft, and to change the y-separation from 33.75 ft (0.9*wingspan) to 29.45 ft (wingspan*π/4). Simulation results are shown for lead and wing UAV velocity, angle-of-attack, sideslip angle, body axis rates, heading, flight path, and velocity roll angles.

Figure 2 shows the lead and wing UAV angles of attack, sideslip angles, and flight speeds. The wing UAV's angle of attack is lower because of the induced lift from the vortex of the lead vehicle. Although it is not shown, the required thrust for the wing vehicle is also much smaller than that required for the lead vehicle. The wing has an initial increase in flight speed due to the commanded change in relative x-separation. Sideslip angles are kept small for both vehicles. Figure 3 shows the μ, γ, χ response of the vehicles. The responses for the lead and wing vehicles are very similar, and both track the desired commands well. Figure 4 shows the body rate response of the vehicles. Here there is a clear difference between the responses. The lead vehicle exhibits a smooth response and is easy to control. The wing vehicle, however, has a much more oscillatory response, demonstrating the difficulty of controlling the wing vehicle as it maneuvers inside the wave vortex of the lead vehicle. The response is still acceptable, however, and the necessary stability and outer-loop command tracking is achieved. Figure 5 shows the relative X, Y, Z separations between the two vehicles, in the reference frame defined by the wing UAV's wind axis frame with zero velocity roll angle. The desired separation commands are tracked effectively, including the variation in the y, z commands to compensate for roll angle, as given by (15). If this modification is left out the x-, y-, z-responses appear very smooth in tracking the resulting constant x-, y-, z-commands.

Further Remarks

Several points should be briefly mentioned for completeness. An adaptive neural network could easily be added to the lead controller's dynamic inversion inner loop. This was not done in this paper due to time constraints , but would improve the vehicle response in the presence of disturbances or more severe dynamic uncertainties.

The adaptive neural network is not in fact required for the wing vehicle to maintain stability and command tracking in the wake of the lead vehicle. The wing vehicle exhibited a reasonable response without the adaptive neural network. However, the response is superior with the adaptive network, and would be much better with increased inversion error.

Control surface deflections are not plotted to save space, and because they do not show anything of interest. The relatively slow outer-loop maneuvers do not drive the inner loop hard enough to make actuator position or rate saturation an issue for the maneuver presented.

5. Conclusions and Future Work

This paper presented a formation flight controls system for UAVs that consisted of an inner-loop adaptive dynamic inversion controller and an outer-loop sliding mode controller. This controller was used in the position control of one UAV flying in the wake of a second UAV. The adaptive inner-loop dynamic inversion controller achieved excellent tracking of p,q,r commands in the presence of large nonlinear formation flight effects. The overall controller performed well, enabling the trailing UAV to achieve and maintain desired positions in the wake of the lead UAV, and to maintain formation while the lead vehicle was maneuvering.

There are many promising directions for future work in this area. Output feedback with realistic measurement errors need to be examined. Another important extension of this work will be to apply it to formations of three or more aircraft, so that string stability issues can be examined. More detailed models of the formation flight vortex effects are also needed. The control methodology developed here can be used for formation joining and separation maneuvers. Both the modeling and control work discussed in this paper also have application to aerial refueling problems for unmanned vehicles, and this area should also be pursued.

References

1. Proud, A., Pachter, M., and D'Azzo, J. J., "Close Formation Control," Proceedings of the AIAA Guidance, Navigation, and Control Conference, AIAA – 99 – 4207, pp. 1231-1246, Portland, OR, August 1999.

2. Wolfe, J.D., Chichka, D.F., and Speyer, J.L., "Decentralized Controllers for Unmanned Aerial Vehicle Formation Flight," in Proceedings of the AIAA Guidance, Navigation, and Control Conference, San Diego, CA, July 1996, 29-31.

3. Chichka, D.F., Park, C.G., and Speyer, J.L., "Peak-Seeking Control with Application to Formation Flight," in Proceedings of IEE Conference on Decision and Control, December 1999.

4. Myatt, J.H., and Blake, W.B., "Aerodynamic Database issues for Modeling Close Formation Flight, in Proceedings of the AIAA Modeling and Simulation Technologies Conference, Portland, OR, August 1999.

5. Singh, S. N., Pachter, M., Chandler, P., Banda, S., Rasmussen, S., Schumacher, C., "Input-Output Invertibility and Sliding Mode Control for Close Formation Flying of Multiple UAVs," AIAA Guidance, Navigation, and Control Conference, AIAA – 00,, Denver CO, August 2000.

6. Kim, B.S. and A.J. Calise, Nonlinear flight control using neural networks. Journal of Guidance, Control, and Dynamics, 1997. 20(1): p. 26-33.

7. Leitner, J., Calise, A., and Prasad, J.V.R., Analysis of Adaptive Neural Networks for Helicopter Flight Control, Journal of Guidance, Control, and Dynamics, 1997, 20(5): p. 972-979.

8. Calise, A.J., Rysdyk, R.T., Nonlinear Adaptive Flight Control Using Neural Networks, IEEE Control Systems Magazine, Dec 1998, V 18 n 6.

9. Schumacher, C., "Adaptive Flight Control Using Dynamic Inversion and Neural Networks," in Proceedings of the 1999 AIAA Guidance, Navigation, and Control Conference.

10. Reiner, J., G.J. Balas, and W.L. Garrard, Robust dynamic inversion for control of highly maneuverable aircraft. Journal of Guidance, Control, and Dynamics, 1995. 18(1): p. 18-24.

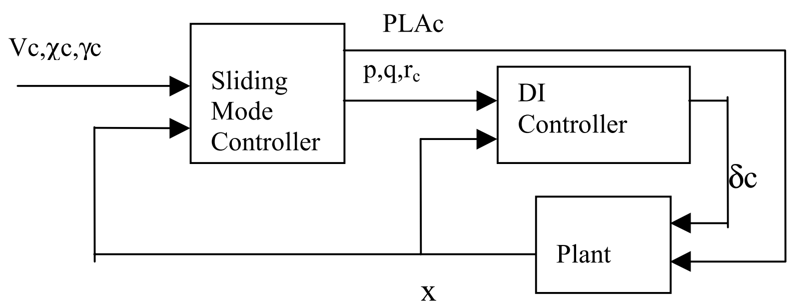

Figure 1: Lead Vehicle Controller Architecture

Figures:

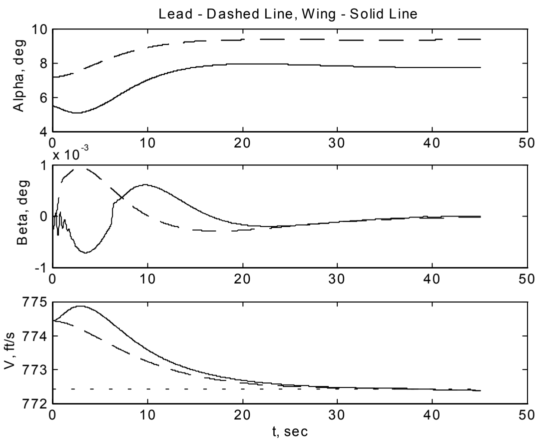

Figure 2 – α, β, Velocity Command Response

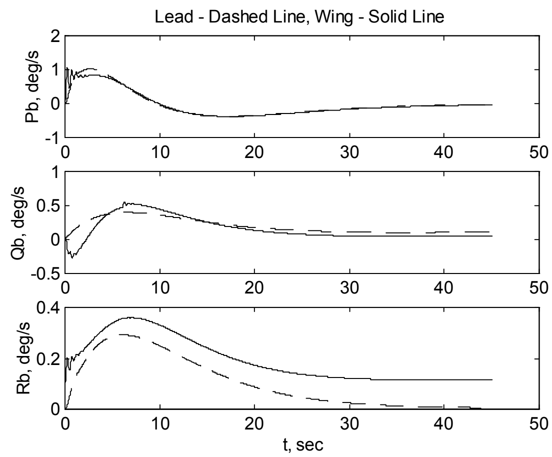

Figure 4 – P, Q, R Response

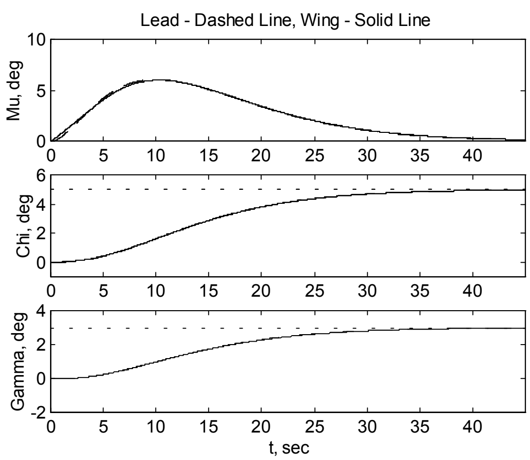

Figure 3 – μ, γ, χ Response

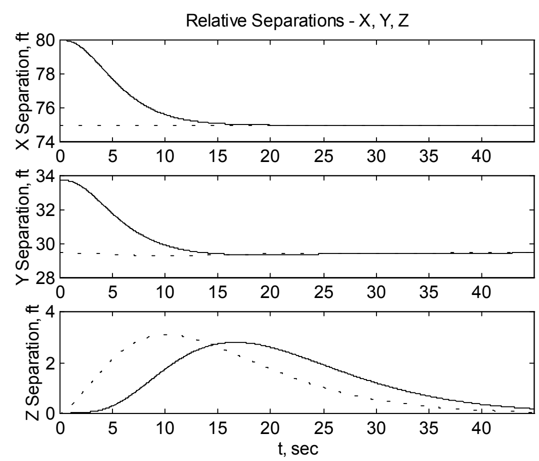

Figure 5 – Relative Separations between UAVs